

Ruby, Rails, and Widgets: Beginning the Future

Brian DeLacey
MIT/BCS
MacTechGroup

Meeting Date: January 3, 2007
Slides revised: January 5, 2007, version 1

Agenda

- Inspiration Starts Here
- The Big Questions
 - History and the GUI'd Future
- Opening Up New Software Frontiers
- Ruby
- Ruby on Rails (RoR)
- Hands-on
- Q&A

Don't wait: Inspiration Starts Here

- **Rails** (this 15 minute video could change your life)
 - http://media.rubyonrails.org/video/rails_take2_with_sound.mov
- **Ruby** (click *Ruby in 20 minutes* for a joyful experience)
 - <http://www.ruby-lang.org/en/documentation/quickstart/>
- **wxRuby2** (get a sense of what a GUI toolkit is like)
 - http://wxruby.rubyforge.org/wiki/wiki.pl?Using_Dialogs_To_Talk_To_Users

The Big Questions

- What will software look like in the future?
- What GUI will people interact with?
- What platforms and tools will dominate?
- What improvements have been made, and will be made, to
 - 1) accelerate development?
 - 2) improve quality?
 - 3) reduce cost of developing/delivering software?

My History

- Programming 1.0 Paradigm
 - Mainframes
 - Minis
 - Micros
 - Macs
 - PCs
 - 1-2-3/Mac
- Programming 1.5 Paradigm
 - Improv
 - Notes
- Plus some Business School Research...
- Where does this lead me to? Web 2.0 and Programming 2.0!

GUI History: Hardware Bindings

- Mac Family Tree
 - Apple I/II/III
 - Alto -> Xerox Star with Mesa POOP (pseudo object-oriented), Tajo IDE, WYSIWYG
 - Lisa
 - Macintosh
- PC Family Tree
 - CP/M-86
 - DR-DOS
 - PC-DOS
 - Windows
- Unix Family Tree
 - X
 - Motif
 - KDE
 - Gnome
 - SunOS4 / NeWS / OpenLOOK / Athena / CDE / Open Look

GUI Toolkits Set Free

- Multi-platform GUI Frameworks
 - wxWidgets
 - FLTK
 - Fox
 - GNU Image Manipulation Program
 - GIMP Toolkit (GTK+)
 - Qt / KDE
- The coming software revolution...
- A time of tremendous change
 - Web 2.0

Opening Up New Software Frontiers

Open Reflections: A Paradigm Shift?

- Stallman marches on Lotus
 - 1989: Radical? *Open source will eat my lunch?*
 - 2009: Visionary! *Open source is my meal ticket!*
- Linux emerges
- LAMP lights the way
- Open Source is everywhere
- Ubuntu: Useful Unix for the rest of us?
- Where have all the MIS operations gone?

Hypotheses: GUI's good-looking future

- The future GUI has browser DNA
 - E.g. Google Apps
- The future GUI works on the web AND desktop
 - You can easily save/retrieve files at either location
- The future GUI is platform independent
 - End users have the same experience independent of underlying microprocessor architecture or run location
 - Multiple device types are naturally supported
- The future GUI is extensible by plug-in libraries
 - E.g. User Communities play increasing developer role
- Here's why ...

Web 2.0

Web 1.0		Web 2.0	
DoubleClick	-->	Google AdSense	
Ofoto	-->	Flickr	
Akamai	-->	BitTorrent	
mp3.com	-->	Napster	
Britannica Online	-->	Wikipedia	
personal websites	-->	blogging	
evite	-->	upcoming.org and EVDB	
domain name speculation	-->	search engine optimization	
page views	-->	cost per click	
screen scraping	-->	web services	
publishing	-->	participation	
content management systems	-->	wikis	
directories (taxonomy)	-->	tagging ("folksonomy")	
stickiness	-->	syndication	

Source: What is Web 2.0, Tim O'Reilly, www.oreilly.net, 9/30/2005

Programming 2.0 Paradigm Shift

Programming 1.0		Programming 2.0	
Compile	-->	Plug-in	
Write->Compile->Run->Test	-->	Write->Test->Run	
Periodic Releases	-->	Real-time updates	
End-User Beta Tests	-->	User Contributed Libraries	
Vendor Controlled	-->	End-User Controlled	
Slow	-->	Fast	
Centralized / Proprietary	-->	Distributed / Open	
Emphasize Production	-->	Emphasize Creation	
Procedural	-->	Object-Oriented	
Custom Code	-->	Customizable Code	
Fragile	-->	Agile	
Monolithic	-->	Modular	
Advantage: Big	-->	Advantage: Small	

Programming 2.0 (continued)

- What is the best way to develop?
 - Desktop programming
 - GUI toolkits?
 - Web programming
 - Frameworks and standards?
- Programming 2.0 = innovative, creative, interactive knowledge creation/application
- This is key to future business IT services; outsourcing this far from HQ is inefficient.

Desktop Programming

- C, C++, C#, Basic (Assembler!) etc...
- GUI toolkit choices
 - Tcl/Tk is included
 - “wxWidgets is a first-class open source response to the need for portability...” - Mitch Kapor
 - “Cross-platform GUI Programming with wxWidgets”
 - wxWidgets major new release for Ruby in 2006
- Where does Ruby fit?

Web Programming

- “I use Python” – Tim Berners- Lee
 - Personal discussion in front of JFK School of Government / Cambridge, circa 1999. I tried Python – and liked it – but kept looking.
- How about the CMS called Zope? (Nope – not for me).
 - While experimenting with Python, I discovered Ruby
- Java – big, powerful, but complex for personal use?
- JavaScript – but I can’t save files locally!
- ActionScript – It’s now open source... (better for Flash?)
- Flex – Novel, new and tempting...
- PHP – ubiquitous on the web; absent on the desktop
- So many options, so little time
- Where does Rails fit?
 - Big projects with small teams; performance and productivity

Is This GUI’d Enough For You?

- Open Source
- Cross Platform
- Elegant, Interactive, Responsive
- International – access to information everywhere
- That was then:
 - “LAMP” (Linux, Apache, MySQL, Python/PHP)
- This is now:
 - “RAMBLE” (Ruby/Rails, Apache/Ajax, MySQL, Browser, Linux, Eclipse) – Web 2.0 is a brighter light than LAMP!

Pause...

- The Desktop / Web programming dilemma
 - Resolution = Ruby + Rails
- I settled on Ruby and Rails as the way...
- ...as my best way to build financial and investment management applications
- Yes ... it's GUI'd enough for me

Ruby Introduction

What is Ruby?

- A language: Pure object-oriented, elegant, powerful, open source programming language developed in Japan in the 90s
 - For years it was lovingly nurtured to cult status in Japan, but remained a captive to its Japanese language. Well-written books appearing in English led to rapid adoption
 - Ruby architect Yukihiro 'Matz' Matsumoto designed this with his favorite languages in mind – Perl, Smalltalk, Eiffel, Ada, and Lisp
 - More details at www.ruby-lang.org
- A community: Friendly, creative, comprehensive, community has contributed libraries, extensions, utilities, documentation, training, and support.

What Kind of Language is Ruby?

- Agile
 - Individual Interactions over processes and tools
 - Working Software over comprehensive documentation
 - Customer collaboration over contract negotiation
 - Responding to change over following a plan
- Learn more
 - http://en.wikipedia.org/wiki/Ruby_%28programming_language%29

Source: agilemanifesto.org

Why Do People Love Ruby?

- Easy and intuitive to use
- Interactive
- Powerful, plentiful libraries, easy C interfaces
- It's a "gem" to install / update (RubyGems)
- Stable and Robust
- A friendly, growing, international community
- Open Source and Multi-Platform
- It's fun to work with
- Try it, you'll like it

Ruby Makes the Mainstream

TIOBE Programming Community Index for January 2007
Ruby declared TIOBE's Programming Language of 2006!

Position Jan 2007	Position Jan 2006	Delta in Position	Programming Language	Ratings Jan 2007	Delta Jan 2006	Status
1	1	=	Java	19.160%	-3.10%	A
2	2	=	C	15.807%	-3.20%	A
3	3	=	C++	10.425%	-1.04%	A
4	5	↑	(Visual) Basic	9.123%	+0.03%	A
5	4	↓	PHP	7.943%	-1.46%	A
6	6	=	Perl	6.237%	-0.81%	A
7	7	=	C#	3.521%	-0.03%	A
8	8	=	Python	3.502%	+0.90%	A
9	10	↑	JavaScript	2.845%	+1.31%	A
10	21	11 * ↑	Ruby	2.519%	+2.15%	A

Source: TIOBE Software, <http://www.tiobe.com/tpci.htm>

Ruby on Rails (RoR) Introduction

What is Ruby on Rails (RoR)?

- Full-stack, open source, Model View Controller (MVC) framework for web apps*
 - Model = Data representation from real world
 - View = User interface presentation, via browser
 - Controller = Algorithmic, programmatic linkage between M/V
- From real-life projects by David Heinemeier Hansson (aka DHH)
 - Written in Ruby, for Ruby, by Rubyists
 - Full-scale integration of embedded Ruby processing on web servers
- Architectural Design
 - Keep it DRY – ‘Don’t Repeat Yourself’
 - ‘Frameworks are Abstractions’ – extracted from live projects at 37Signals
 - ‘Convention over Configuration’ – a standards view of software engineering
 - MetaProgramming - Program generated code = 44 files in 36 directories as skeleton app

* The MVC architectural pattern/philosophy for developing software systems was developed by Trygve Reenskaug, in 1979 while working on Smalltalk at Xerox. (A descriptive paper is <http://st-www.cs.uiuc.edu/users/smarch/st-docs/mvc.html>.)

What's This Rails Buzz About?

- Productivity for the Developer
 - Simplified development and maintenance
- Open Source: MIT License!
- Scaffolding, Templates
 - E.g. No need to define database column names, Ruby knows
- ActiveRecord – The Zen of CRUD
 - Object-relational mapping (ORM) made easy
- Ajax support – part of the framework
- Plug-ins Contributed by Enthusiastic Rails community
- One-Click Installer for Windows runs web server on local host
- Friendly comparisons/competition with Python, PHP ...

Did I Mention Ajax and Rails?

- Rails supports easy implementation of Ajaxian features
 - Prototype wraps XMLHttpRequest
 - Script.aculo.us builds on top of Prototype
 - Nice GUI features e.g. drag-and-drop
- What more could you ask for?

Development Environment = Free

- Ruby
 - <http://www.ruby-lang.org/en/downloads/>
- Rails
 - <http://www.rubyonrails.org/down>
- Eclipse
- Putty
- Nano
- FTP
- Local Linux / Ubuntu Box running RAMBLE
-

Hosting (Costs Just a Little Extra)

- Install Ubuntu – give life to an aging PC
 - Install Rails, Ruby and set up your own test net
- Find an ISP / Server with Ruby on Rails support
 - DreamHost is great at <http://www.dreamhost.com/>
 - Super value (ie. Inexpensive)
 - Unlimited databases
 - Great storage and bandwidth
 - Self Promotion Aside
 - Promocode 'greaterboston' saves \$50
 - on 1 and 2 year accounts
 - there are many promo codes out there ... some save more

Rails Sites and Resources

- <http://www.37signals.com/>
 - <http://www.basecamphq.com/tour>
- <http://www.43things.com/>
- Learn More
 - <http://www.rubycentral.com/book/intro.html>
 - Greater Boston Ruby and Rails Group
 - See more detailed Ruby/Rails notes at
 - www.greaterbostonrubyandrails.com

Rails Recap

- Amazing Traction and Accelerating Adoption
 - Initially released in July 2004
 - Version 1.0 released in December 2005
 - Has taken off like a rocket

A Little R&R in the Future?

- Ruby – Bytecode compiler
 - End of 2007?
- Rails – ActiveRecord
 - Already gaining traction
- This area is moving VERY fast...
- Expect the next innovation
 - from someplace you've never been to
 - from someone you've never heard of

Hands-On, Q&A

Let's Look Under the Hood

- Ruby Sample Code
 - Examples
 - Eclipse – how it works, why it is useful
- wxRuby2
 - Sample Code
 - Yes, you can compile this for Windows!
- Rails Sample Code
 - Server editing: make changes and see results

Possible Future Meeting Topics

- Intro - Ruby and Rails
- Intermediate Rails – Building Websites
- Advanced Rails – Deploying Applications
- Intermediate Ruby – Deep Dive Programming
- Advanced Ruby - Libraries
- Fun Hacks for Ruby and Rails
- Making the Switch to Ruby (from Java, Python ...)
- Magical and Mystical Tools (Regular Expressions, Ajax ...)
- History of Open Source, Ruby, and Rails
- OS Platforms – Linux/Ubuntu, Mac, Windows
- Cutting Edge – Software Design and Development
- Hardware Hacking – Performance Unlimited
- Enterprise Ruby and Rails

Thank you! Q&A

- If you'd like to join the *Greater Boston Ruby and Rails User Group*
 - Send an email to bdelacey@gmail.com
 - Visit www.greaterbostonrubyandrails.com